Hướng dẫn chi tiết về Docker cho Ollama và FastAPI

Mục lục

Tổng quan

Dự án sử dụng Docker để containerize hai service chính:

1. **FastAPI backend service**
2. **Ollama LLM service**

Cấu trúc Docker

1. FastAPI Dockerfile

```
{ Alex Dev }

FROM python:3.11-slim

WORKDIR /app
```

```
COPY ./requirements.txt /app/requirements.txt
RUN pip3 install -r /app/requirements.txt

CMD ["uvicorn", "app:app", "--host", "0.0.0.0", "--port", "8000", "--
reload"]
```

Giải thích từng phần:

- `python:3.11-slim`: Base image nhẹ với Python 3.11
- `WORKDIR /app`: Thiết lập thư mục làm việc
- `COPY ./requirements.txt`: Copy file dependencies
- `RUN pip3 install`: Cài đặt các dependencies
- `CMD ["uvicorn"]`: Khởi chạy FastAPI server với Uvicorn

2. Ollama Dockerfile

```
{ Alex Dev }

FROM ollama/ollama:latest

# Expose port 11434 for API access
EXPOSE 11434

# Set environment variables
ENV OLLAMA_HOST=0.0.0.0
ENV OLLAMA_ORIGINS=*

# Pull model during build
RUN ollama pull qwen2.5-coder:0.5b

# Start Ollama server
CMD ["ollama", "serve"]
```

Giải thích từng phần:

- `ollama/ollama:latest`: Base image chính thức từ Ollama
- `EXPOSE 11434`: Port cho Ollama API
- `ENV OLLAMA_HOST`: Cho phép truy cập từ bên ngoài
- `ENV OLLAMA_ORIGINS`: CORS configuration
- `RUN ollama pull`: Tải model trong quá trình build
- `CMD ["ollama", "serve"]`: Khởi chạy Ollama server

Docker Compose

{ Alex Dev }

```
version: '3.8'

services:
  fastapi:
    build:
      context: ./fastapi
      dockerfile: Dockerfile
    ports:
      - "8000:8000"
    volumes:
      - ./fastapi:/app
    depends_on:
      - ollama
    networks:
      - app-network

  ollama:
    build:
      context: ./ollama
      dockerfile: Dockerfile
    ports:
      - "11434:11434"
```

```
    volumes:
      - ollama_data:/root/.ollama
    networks:
      - app-network
    deploy:
      resources:
        reservations:
          devices:
            - driver: nvidia
              count: 1
              capabilities: [gpu]

volumes:
  ollama_data:

networks:
  app-network:
    driver: bridge
```

Giải thích cấu hình:

1. **FastAPI Service:**

- **Port mapping: 8000:8000**
- **Volume mount cho hot reload**
- **Phụ thuộc vào Ollama service**
- **Kết nối qua app-network**

2. **Ollama Service:**

- **Port mapping: 11434:11434**
- **Volume cho model storage**
- **GPU support qua NVIDIA driver**
- **Kết nối qua app-network**

Các lưu ý quan trọng

1. Network Configuration

```
{ Alex Dev }

url_docker = "http://ollama-server:11434"
#test on local
url_local = "http://localhost:11434"
```

- `url_docker`: Sử dụng hostname `ollama-server` khi chạy trong Docker
- `url_local`: Sử dụng `localhost` khi chạy locally

2. Volume Management

- FastAPI volume mount cho development
- Ollama volume cho model persistence

3. Resource Management

- GPU allocation cho Ollama
- Memory limits nên được cấu hình

Best Practices

1. **Security:**

- Sử dụng non-root user trong containers
- Limit container capabilities
- Scan images cho vulnerabilities

2. **Performance:**

{ Alex Dev }

```yaml
services:
  ollama:
    deploy:
      resources:
        limits:
          memory: 8G
        reservations:
          memory: 4G
```

3. **Development Workflow:**

- Sử dụng multi-stage builds
- Implement health checks
- Optimize layer caching

4. **Monitoring:**

{ Alex Dev }

```yaml
services:
  fastapi:
    healthcheck:
      test: ["CMD", "curl", "-f", "http://localhost:8000/health"]
      interval: 30s
      timeout: 10s
      retries: 3
```

5. **Environment Variables:**

{ Alex Dev }

```
services:
  fastapi:
    env_file:
      - .env
    environment:
      - OLLAMA_API_URL=http://ollama:11434
      - LOG_LEVEL=info
```

Tài liệu tham khảo

1. Docker Documentation
2. FastAPI in Containers
3. Ollama Docker Guide
4. Docker Compose Networking